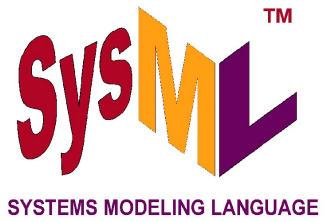


Systems Modeling Language™ Overview

Cris Kobryn and Sandy Friedenthal

SysML Partners
(www.SysML.org)

October 2003

Overview

- Background
- Charter
- Partners
- Requirements & Evaluation Criteria
- Design Goals
- UML 2.0 Reuse
- AP-233 Architectural Alignment
- Deliverables & Milestones

Background

- Systems Engineers need a common specification language for SE applications
- OMG Systems Engineering Domain Special Interest Group
  - joint INCOSE-OMG initiative chartered in 2001 to extend UML to SE applications
  - collaborated with UML2 submission teams
  - drafted UML for SE Request for Proposal, issued by the OMG in March 2003
- SysML Partners organized in May 2003 to respond to UML for SE RFP

Charter

- The SysML Partners are collaborating to define a modeling language for systems engineering applications, called Systems Modeling Language™ (SysML™). SysML will customize UML 2.0 in order to support the specification, analysis, design, verification and validation of complex systems that include hardware and software components.

SysML Partners

- Industry
  - BAE SYSTEMS, Deere & Company, Lockheed Martin, Motorola, Northrop Grumman, Raytheon, Thales
- Government
  - NASA/JPL, NIST, OSD
- Tool Vendors
  - Artisan, Ceira, IBM/Rational, I-Logix, Telelogic, Vitech
- Liaisons
  - AP-233, INCOSE, Rosetta

Requirements

- Structure
  - e.g., system hierarchies, interconnections
- Behavior
  - e.g., function-based behaviors, state-based behaviors
- Properties
  - e.g., parametric models, time variable attributes
- Requirements
  - e.g., requirements hierarchies, traceability
- Verification
  - e.g., test cases, verification results

Evaluation Criteria

- Ease of use
- Unambiguous
- Precise
- Complete
- Scalable
- Adaptable to different domains
- Capable of complete model interchange
- Evolvable
- Process and method independent
- Compliant with UML 2.0 metamodel
- Verifiable

Design Goals

- Satisfy UML for SE RFP requirements
  - 6.5 Mandatory req’ts, 6.6 Optional req’ts
- Reuse UML 2.0 to the extent practical
  - select subset of UML 2.0 reusable for SE apps
  - parsimoniously add new constructs and diagrams needed for SE
- Incrementally grow the language
  - prevent scope and schedule creep
  - take advantage of SE user feedback as language evolves via minor and major revisions
- Architecturally align SysML with AP-233 SE Data Interchange
- Ensure that SysML scales to support architectural frameworks, such as DODAF/C4ISR

UML 2.0 Reuse & Customization

- SysML will re-use subset of UML 2.0
  - 20% used to do 80% of common modeling work + selected constructs
- SysML will customize UML 2.0 in 2 ways
  - customize graphic nodes and paths (boxes and lines) with names and icons consistent with SE usage (e.g., «system»)
  - add new constructs and diagrams for new capabilities (e.g., parametric diagram)

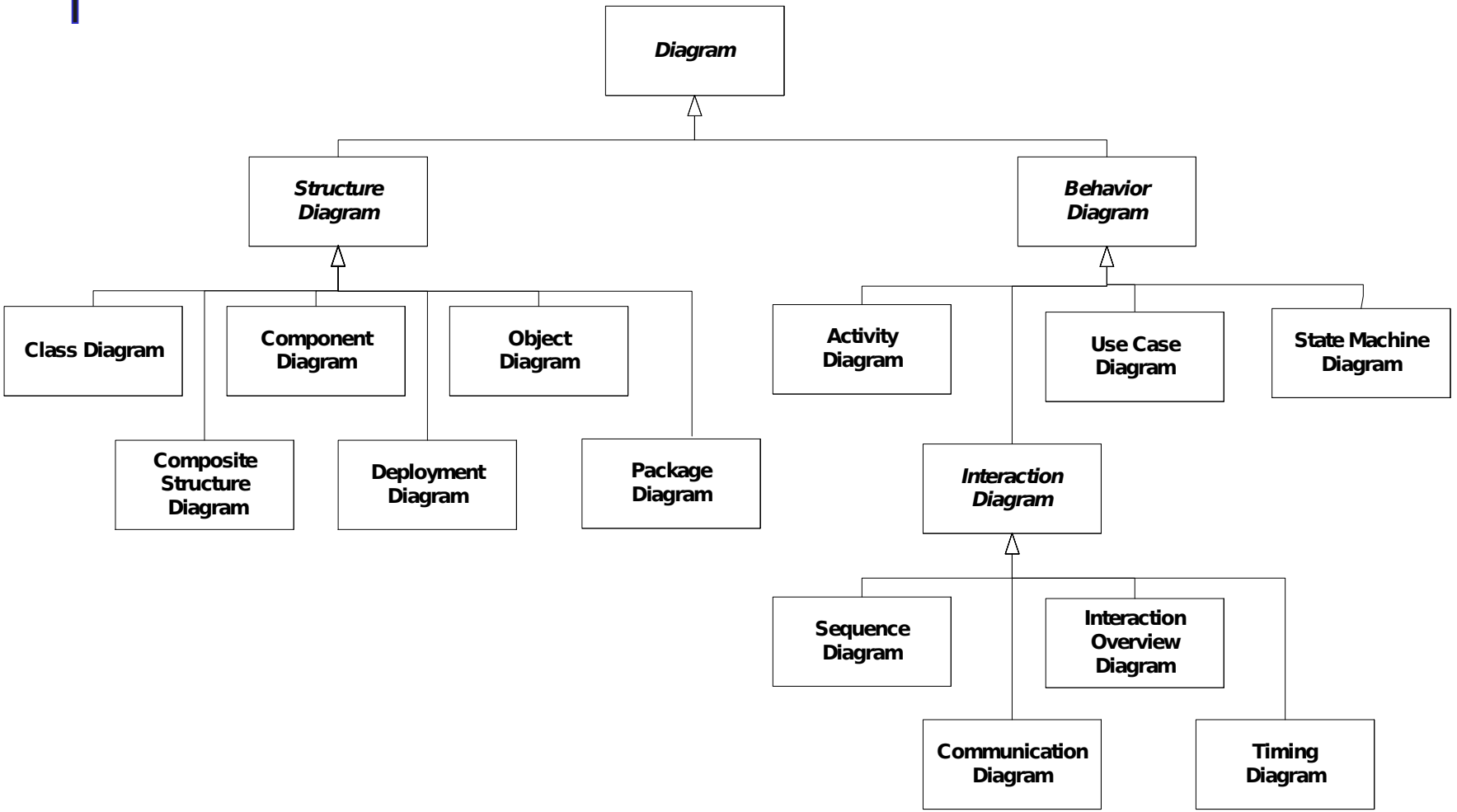
UML 2.0 Support for SE

- Allows for more flexible System, Subsystem and Component representations
- Structural decomposition
  - e.g., Classes, Components, Subsystems
- System and component interconnections
  - via Parts, Ports, Connectors
- Behavior decomposition
  - e.g., Sequences, Activities, State Machines
- Enhancements to Activity diagrams
  - e.g., data and control flow constructs, activity partitions/swim lanes

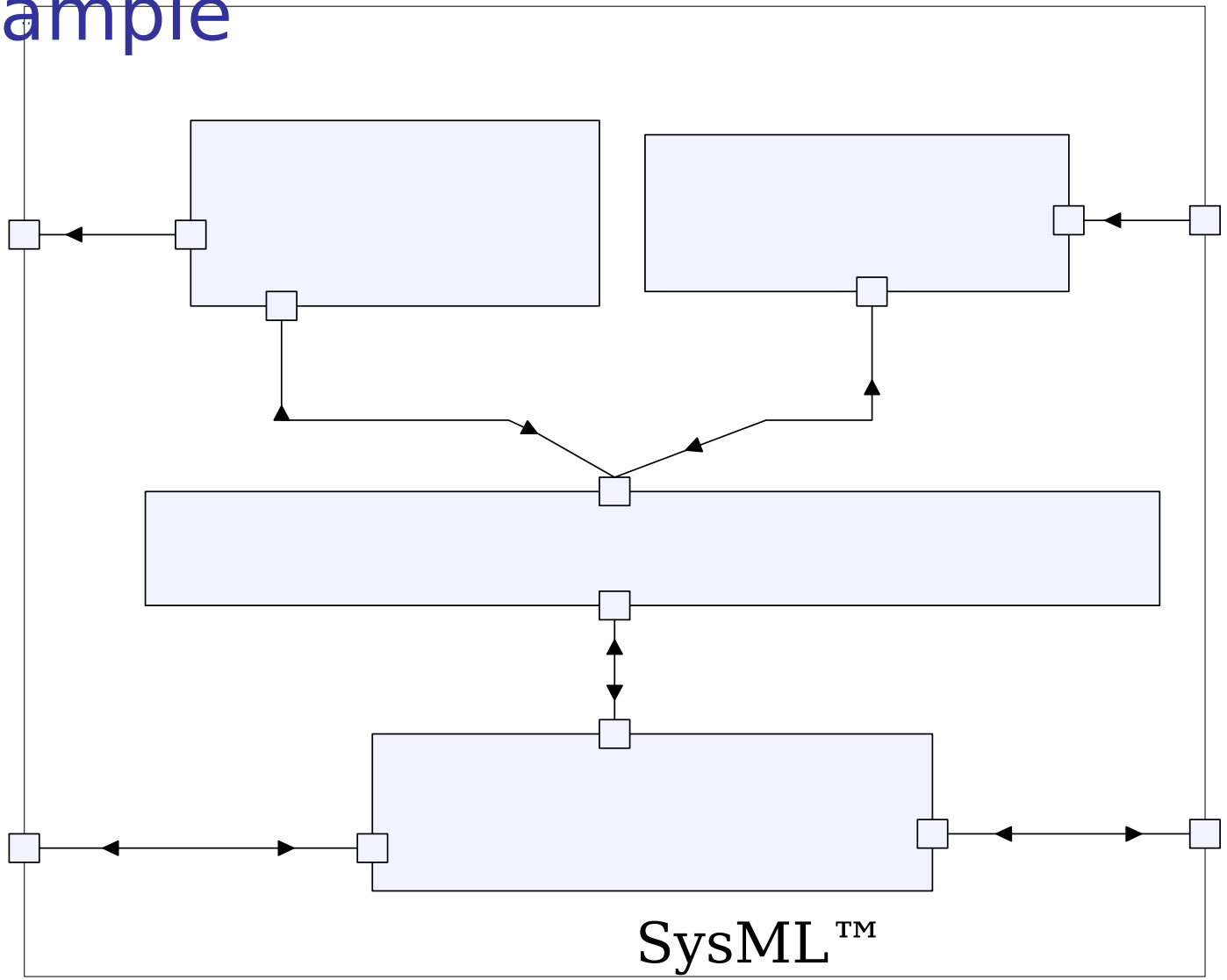
UML 2.0 Support for SE (cont.)

- Enhancements to Interaction diagrams
  - e.g., alternative sequences, reference sequences, interaction overview, timing diagrams
- Support for information flows between components
- Improved Profile and extension mechanisms
- Support for complete model interchange, including diagrams
- Compliance points and levels for standardizing tool compliance
- Does not preclude continuous time varying properties
  - especially important for SE applications

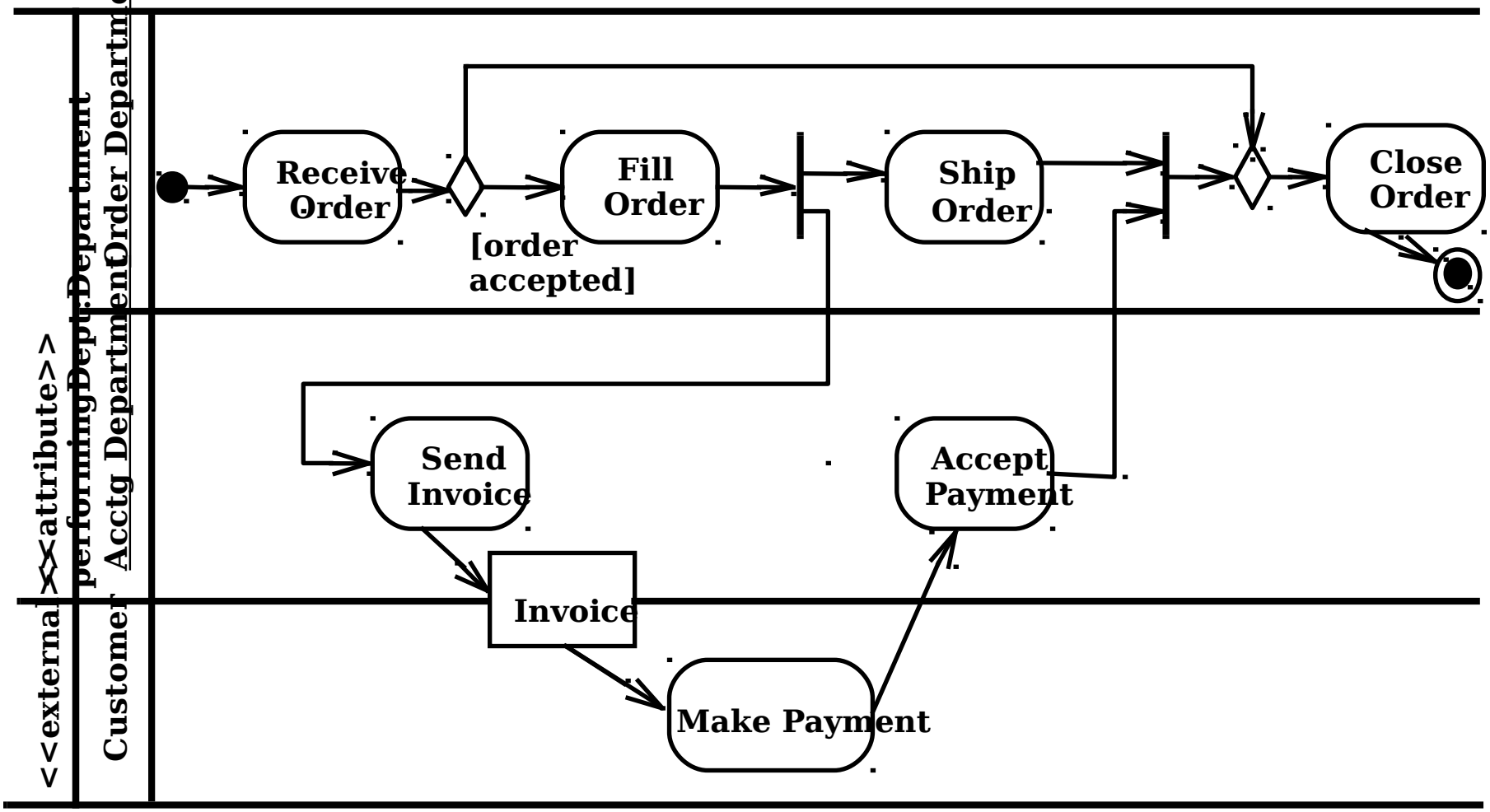
UML 2.0 Diagram Taxonomy



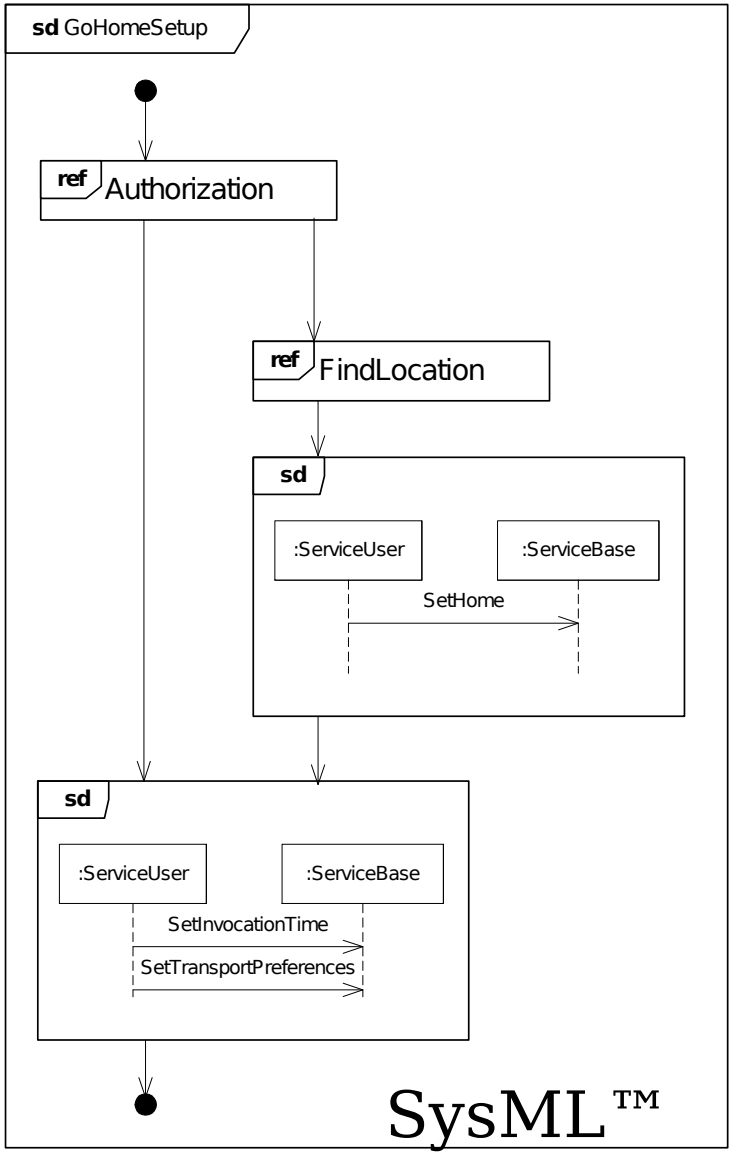
Composite Structure Diagram Example



Activities: Hierarchical Partitions



Interaction Overview Diagram



UML2 for SE Customization Areas

- Requirements constructs
- Parametric Diagrams
- Activity diagram customization to support continuous time
- Other constructs and new diagrams to be identified and prioritized

AP-233 Architectural Alignment

- AP233 Demonstrator Tool
  - facilitate data interchange
- Example Collaboration
  - facilitate metamodel alignment

AP233 Demonstrator Tool

- Tool developed as part of AP233's first phase (Requirements data model)
- Plays two roles:
  - creates sample test data for Requirements and Product Structure
  - performs first order validation of test exchanges
- Tool to be extended as next phase module sets are funded
- Tool available free for non-commercial use

SysML Use of Demonstrator Tool

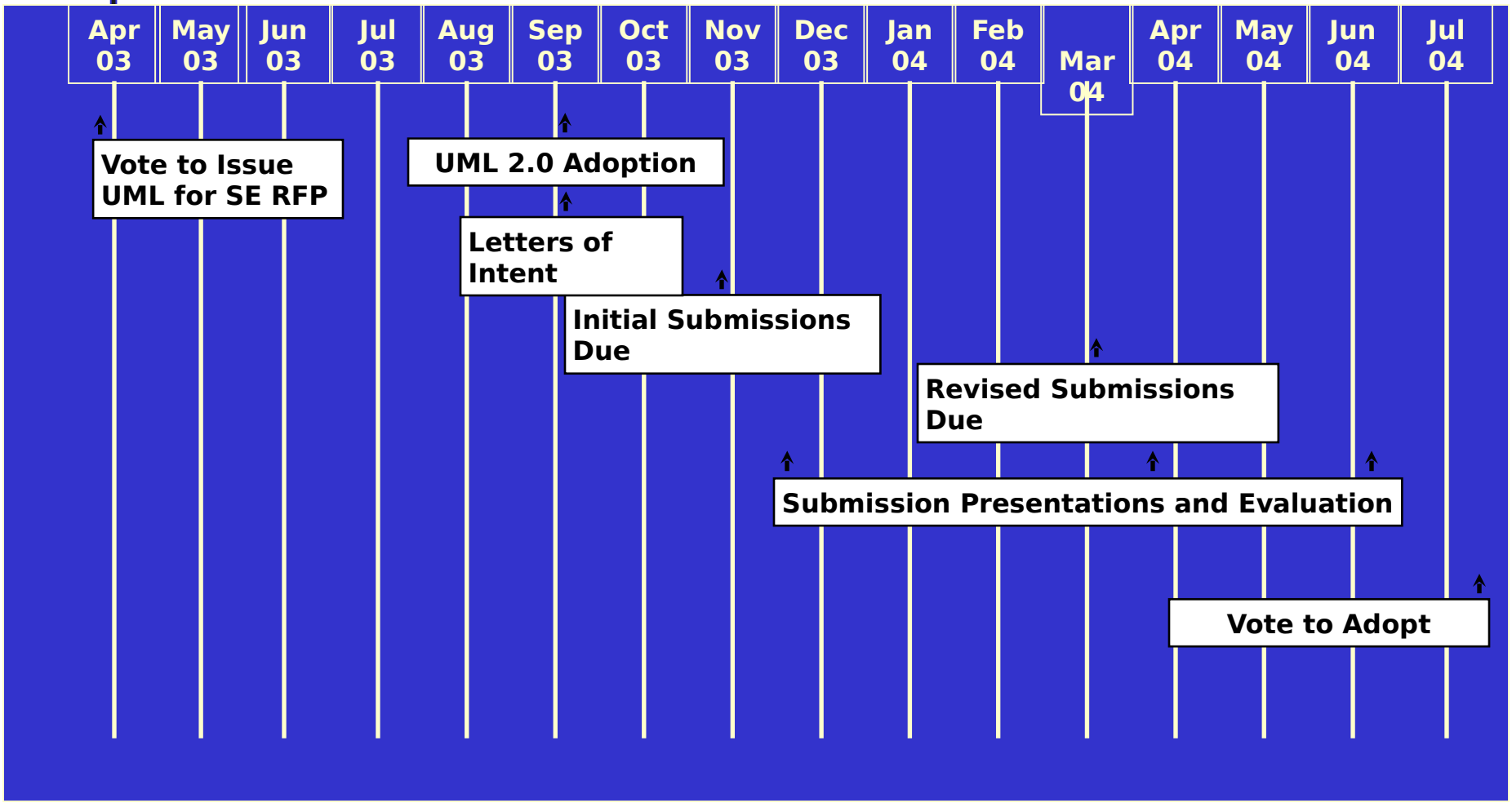
- Can currently be used in prototype testing of XMI interfaces for
  - Requirements
    - Text-based
    - Property-based
  - Product Structure
  - Next phases of AP233 development
    - Structures
    - Behavior

Note: schedule of future phases dependent on funding.

SysML Submission Deliverables

- Language specification as a UML 2.0 profile (customization)
  - optional metamodel
- Sample problems
  - illustrate practical SE applications
  - use with tool prototypes as proof-of-concept
- Full requirements traceability
  - ensure language satisfies SE needs

Milestones


Apr 03
May 03
Jun 03
Jul 03
Aug 03
Sep 03
Oct 03
Nov 03
Dec 03
Jan 04
Feb 04
Mar 04
Apr 04
May 04
Jun 04
Jul 04
Vote to Issue UML for SE RFP
UML 2.0 Adoption
Letters of Intent
Initial Submissions Due
Revised Submissions Due
Submission Presentations and Evaluation
Vote to Adopt

Wrap up

- Systems Engineers need to customize UML 2.0 to specify complex systems
- OMG has issued a UML for SE RFP that captures Systems Engineering requirements
- SysML Partners are specifying the Systems Modeling Language (SysML) to respond to the UML for SE requirements
- SysML will be architecturally aligned with ISO AP-233
- Initial submission due Oct. 2003; revised submission due Mar. 2004

Further Info

- Web
  - www.SysML.org
- Public mailing lists
  - SysML-info@yahoogroups.com
  - SysML-issues@yahoogroups.com
- Chairs
  - Cris Kobryn (cris@ieee.org)
  - Sandy Friedenthal (sanford.friedenthal@lmco.com)